

RAPPORT

SYSTÈME DE PRÉDICTION DU RISQUE DE RÉADMISSION PREDICARE

Realisée par :
TAOUIK DAHAMI
CHAHBAR ALIAA
BEN MOUSSA ANOUAR
AMEZIANE DOUAA

The screenshot shows the PrediCare application interface. At the top left is the logo "PrediCare". On the right side, there is a small profile picture of a woman. The main content area is titled "Détail du patient : Jean Dupont".

- Informations Démographiques:**
 - ID Patient: P0012345
 - Date de Naissance: 15/05/1978
 - Genre: Masculin
 - Téléphone: +33 6 12 34 56 78
 - Adresse: 12, Rue de la Paix, 75001 Paris
 - Email: jean.dupont@example.com
- Médicaments Actuels:**
 - Metformine - 500 mg, 2 fois par jour
 - Valsartan - 80 mg, 1 fois par jour
 - Aspirine - 75 mg, 1 fois par jour
- Antécédents Médicaux:**
 - Diabète de type 2 (Diagnostiqué le 10/03/2010)
 - Hypertension artérielle (Diagnostiquée le 22/07/2015)
 - Allergie à la pénicilline (Diagnostiquée le 01/01/1995)
- Rencontres Récentes:**
 - 20/11/2023: Consultation (Contrôle diabète)
 - 05/09/2023: Examen de laboratoire (Bilan sanguin)
 - 18/07/2023: Urgence (Douleur thoracique)
- Prédiction de Risque de Réadmission:**

Prédire le risque de réadmission

Cliquez sur le bouton ci-dessus pour obtenir la prédiction.

At the bottom center of the page is the copyright notice: © 2025 PrediCare. All rights reserved.

- Fonctionnalité :** Dossier détaillé pour un patient spécifique (e.g., Jean Dupont).
- Sections :**
 - Informations Démographiques (ID, Date de Naissance, Genre, etc.).
 - Antécédents Médicaux (Diabète, Hypertension, Allergie).
 - Médicaments Actuels (Metformine, Valsartan, Aspirine).
 - Rencontres Récentes (Consultation, Examen de laboratoire, Urgence).
- Fonctionnalité Cœur :** Section Prédiction de Risque de Réadmission avec un bouton interactif "Prédire le risque de réadmission" pour déclencher ou consulter la prédiction.

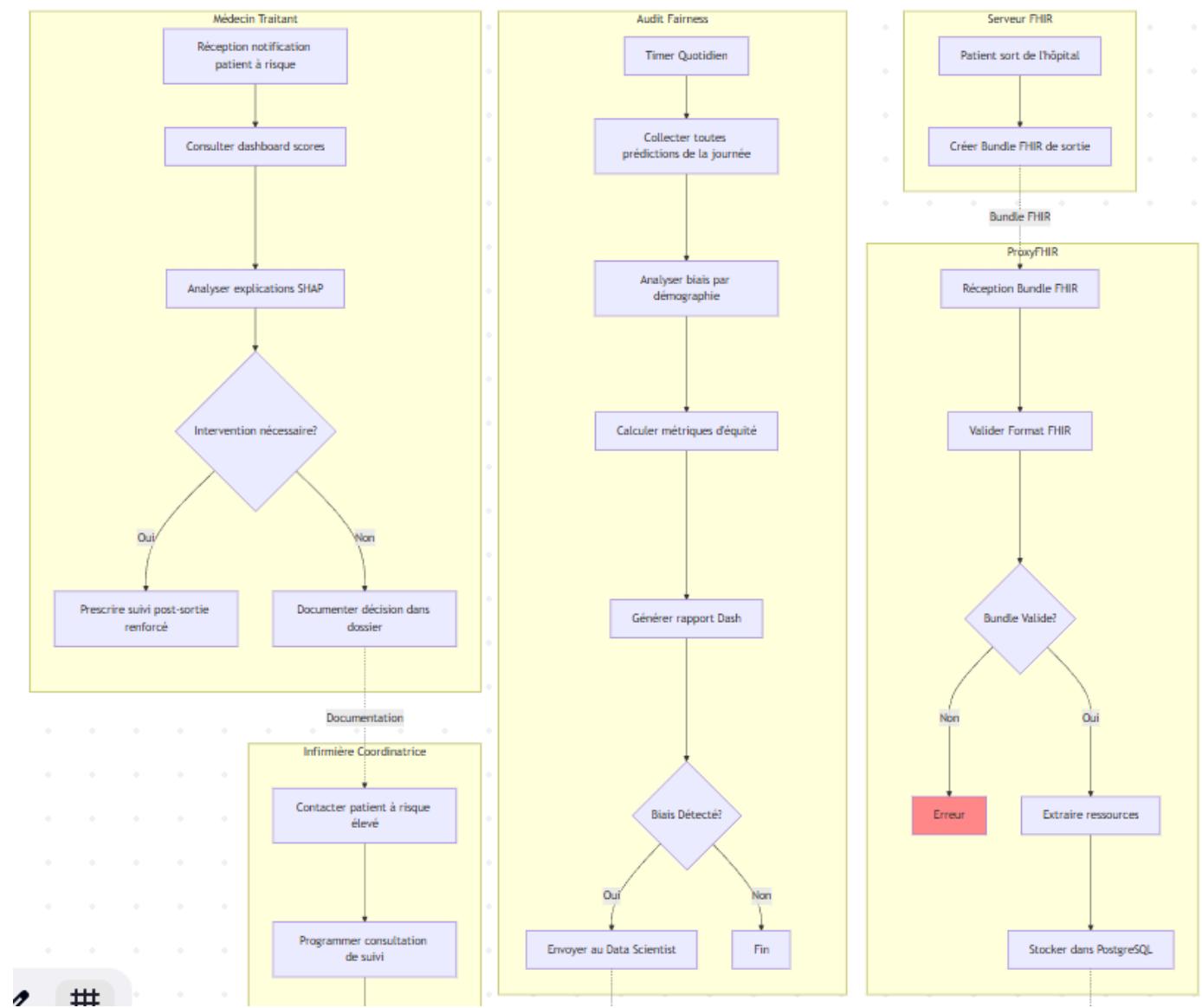
4.4. Configuration des Service

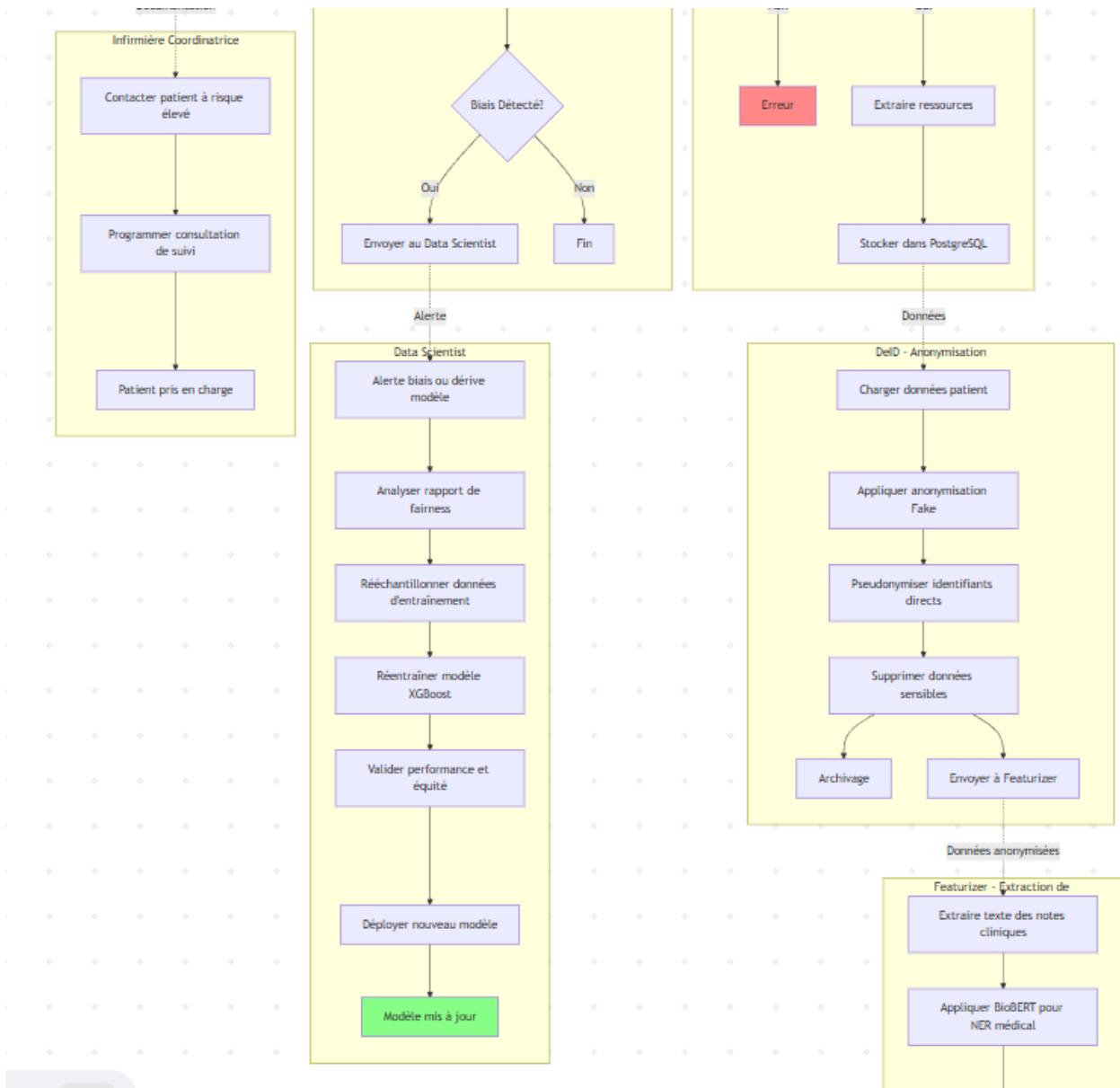
Introduction

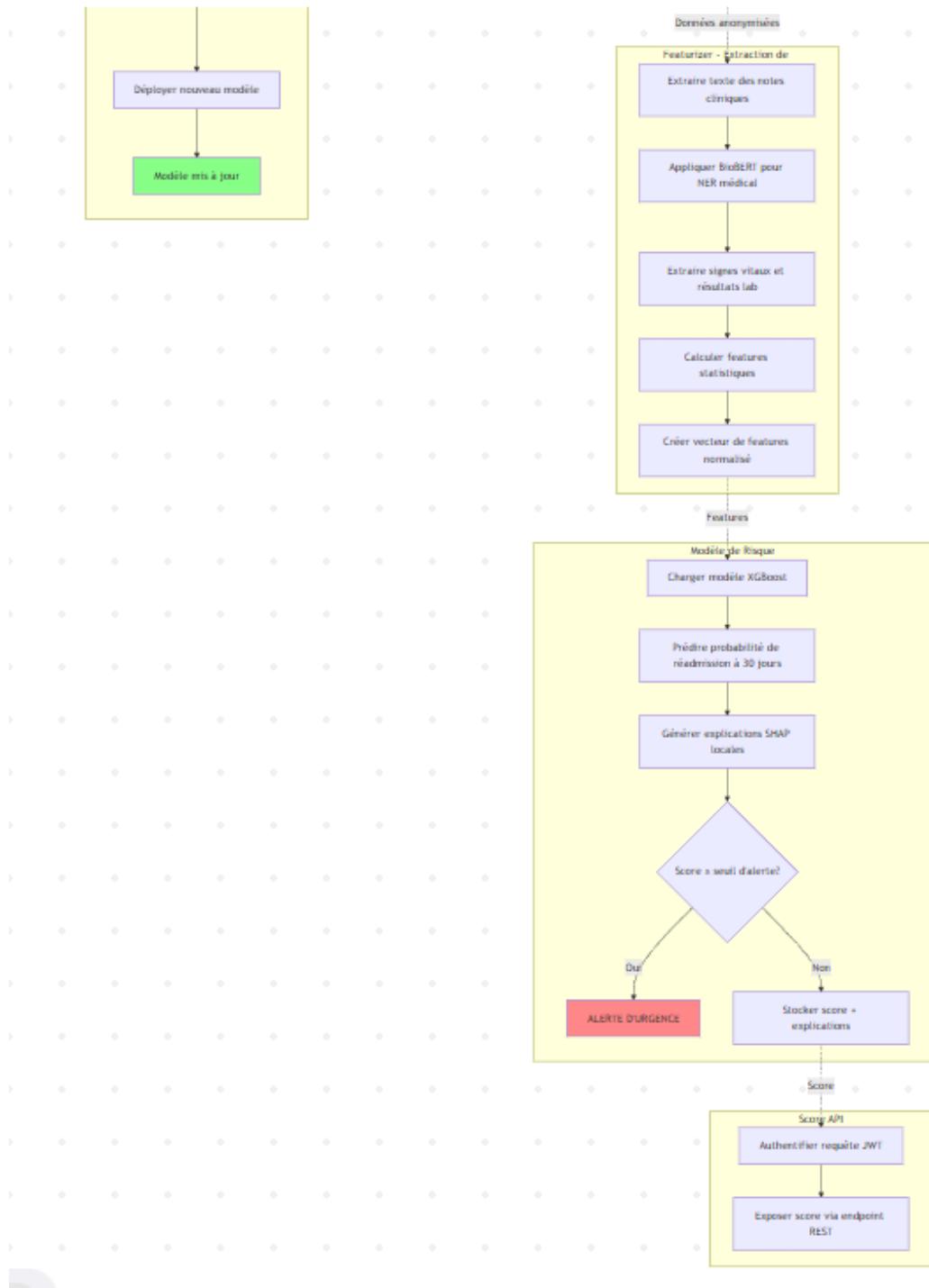
Ce rapport présente l'architecture, la conception et les processus métiers du système PrediCare, une solution basée sur des microservices destinée à prédire le risque de réadmission des patients.

1. Diagramme BPMN avec Description Détailée des Processus Métiers

Le système PrediCare est basé sur plusieurs processus clés couvrant l'ingestion des données, le scoring, la documentation médicale et l'audit de l'équité.







Processus Global

1.1. Ingestion de Données et Traitement Asynchrone (FHIR à PostgreSQL)

Ce processus est déclenché lorsqu'un patient sort de l'hôpital, générant un **Bundle FHIR** de sortie par le **Serveur FHIR** externe.

1. **Réception du Bundle FHIR (ProxyFHIR)** : Le microservice ProxyFHIR reçoit le Bundle FHIR.
2. **Validation FHIR** : Il vérifie si le format du Bundle est valide.
 - o Si **Non** : Un message d'erreur est généré.
 - o Si **Oui** : Les ressources sont extraites et stockées dans la base de données **PostgreSQL**.

1.2. Chaîne de Traitement du Modèle (Scoring et Explication)

Ce processus s'applique aux données stockées dans PostgreSQL et est principalement asynchrone, dirigé par un flux de travail de données :

1. **Anonymisation (DeID)** :
 - o Le microservice **DeID** charge les données patient (PII - Personally Identifiable Information).
 - o Il applique l'anonymisation Falke, pseudo-anonymise les identifiants directs et supprime les données sensibles.
 - o Les données anonymisées sont archivées puis transmises au Featurizer.
2. **Extraction et Transformation (Featurizer)** :
 - o Le microservice **Featurizer** reçoit les données anonymisées.
 - o Il extrait le texte des notes cliniques et applique **BioBERT** pour la reconnaissance d'entités médicales (NER).
 - o Il extrait les signes vitaux et les résultats de laboratoire.
 - o Il calcule des features statistiques et crée un **vecteur de features normalisé**.
3. **Prédiction du Risque (ModelRisque)** :
 - o Le microservice **ModelRisque** charge le modèle **XGBoost** entraîné.
 - o Il prédit la probabilité de réadmission à 30 jours.
 - o Il génère les explications locales **SHAP** pour la prédiction.
4. **Stockage et Exposition du Score (ScoreAPI)** :
 - o Le score et les explications sont stockés dans la base de données PostgreSQL.
 - o Si le score dépasse un seuil critique, une **ALERTE D'URGENCE** est générée.
 - o Le microservice **ScoreAPI** authentifie la requête via **JWT Auth** et expose le score via un endpoint REST.

1.3. Flux Clinique (Médecin Traitant et Infirmière Coordinatrice)

Ce processus est déclenché par un score de risque élevé (synchrone via l'interface web) :

1. **Réception de la notification** : Le **Médecin Traitant** reçoit une notification de patient à

- risque.
2. **Consultation** : Il consulte le dashboard et les scores, puis analyse les explications SHAP fournies par le système.
 3. **Décision** :
 - o Si une **Intervention est Nécessaire** : Il prescrit un suivi post-sortie renforcé.
 - o Si **Non** : Il documente la décision dans le dossier.
 4. **Coordination** : L'**Infirmière Coordinatrice** contacte le patient à risque élevé et programme une consultation de suivi.

1.4. Audit de l'Équité (AuditFairness)

Ce processus est exécuté de manière quotidienne pour moniturer le modèle et détecter d'éventuels biais :

1. **Collecte des Prédictions** : Le microservice **AuditFairness** collecte toutes les prédictions de la journée.
2. **Analyse de Biais** : Il analyse le biais par démographie et calcule les métriques d'équité.
3. **Rapport et Alerte** : Il génère un rapport **Dash** (EvidentlyAI).
 - o Si un **Biais est DéTECTé** : Une alerte est envoyée au Data Scientist pour rééchantillonner les données, ré-entraîner le modèle **XGBoost**, valider sa performance et son équité, et enfin déployer le nouveau modèle.
 - o Si **Non** : Le processus se termine.

2. Architecture Microservices

2.1. Schéma d'Architecture (Vue d'ensemble)



Le système suit une architecture orientée microservices, orchestrée par une API Gateway.

Le flux principal est :

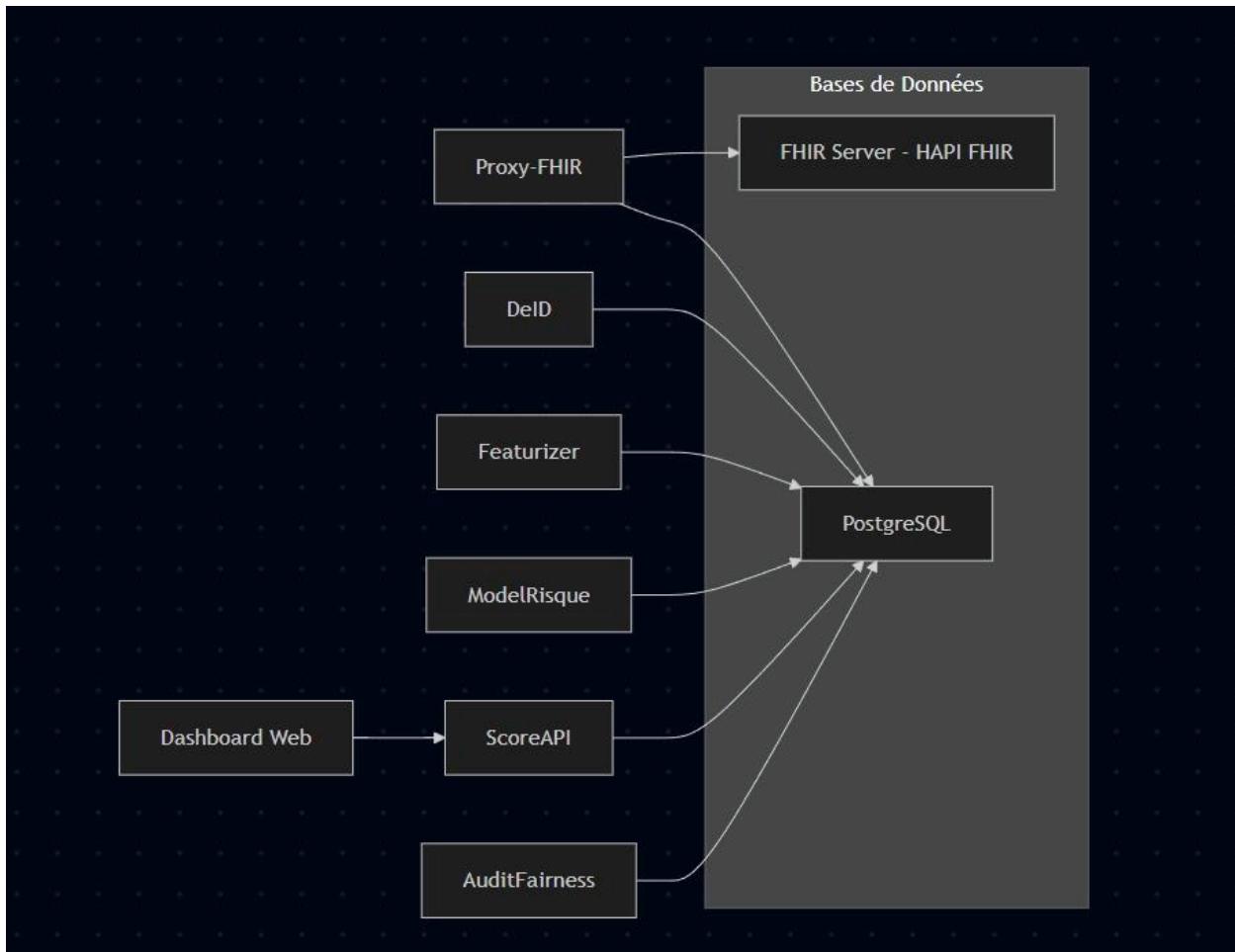
Dashboard Web -> API Gateway-> ProxyFHIR -> DeID -> Featurizer-> ML Predictor
(ModelRisque/ScoreAPI).

2.2. Rôle, Technologies et Base de Données des Microservices

Microservice	Rôle Principal	Technologies Clés	Base de Données Associée

API Gateway	Point d'entrée unique, routage, sécurité, gestion des requêtes.	Non spécifié, mais agit comme un frontal pour les services.	Aucune (ou cache pour la configuration)
ProxyFHIR	Interface avec le serveur FHIR externe, validation, synchronisation des données FHIR.	Java, HAPI FHIR, Spring Boot, Spring Web, Spring Data JPA, PostgreSQL JDBC Driver.	PostgreSQL
DeID	Anonymisation et pseudo-anonymisation des données PII.	Python, Faker.	PostgreSQL
Featurizer	Extraction d'entités médicales (NER) à partir de notes cliniques, création de vecteurs de features.	Python, BioBERT, spaCy.	PostgreSQL
ModelRisque	Chargement du modèle XGBoost, prédiction du risque de réadmission, génération des explications SHAP.	Python, XGBoost, SHAP.	PostgreSQL
ScoreAPI	Exposition du score de risque via une API REST sécurisée (JWT Auth).	FastAPI, JWT Auth.	PostgreSQL
AuditFairness	Surveillance de l'équité du modèle, détection des biais.	EvidentlyAI, Python, Dash.	PostgreSQL

2.3. Base de Données Associée



Tous les microservices métier (ProxyFHIR, DeID, Featurizer, ModelRisque, ScoreAPI, AuditFairness) interagissent avec une base de données centralisée **PostgreSQL**, qui stocke les données des patients extraites du serveur FHIR. Le **FHIR Server (HAPI FHIR)** externe est la source de données initiale.

2.4. Méthodes de Communication entre Microservices

La communication dans le système PrediCare est un mélange d'appels synchrones (via HTTP) pour le flux de requête/réponse immédiat et de flux asynchrones (via la base de données/file d'attente, non spécifié mais implicite pour le traitement lourd).

Communication Synchrone (Exemple pour ProxyFHIR) :

L'appel au serveur FHIR est synchrone et géré par le microservice ProxyFHIR.

- **Outil principal** : Microservice ProxyFHIR (projet proxy-fhir).
- **Endpoint** : FhirProxyController -> endpoint POST /fhir/sync/bulk.
- **Client FHIR** : FhirClientService qui utilise **HAPI FHIR** (IGenericClient) pour appeler le vrai serveur FHIR externe.

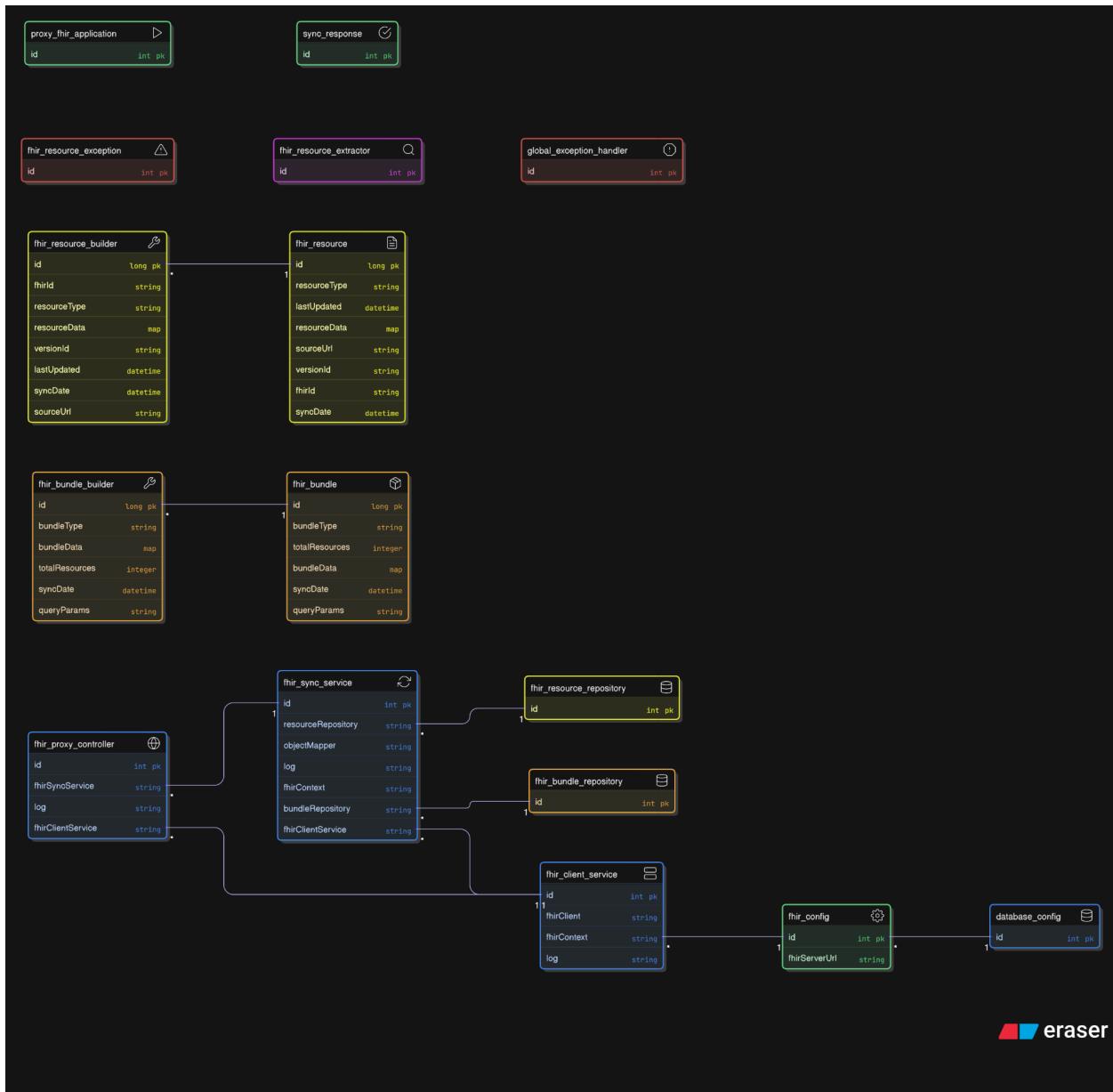
Communication Asynchrone :

Le flux de traitement (DeID -> Featurizer-> ModelRisque) est intrinsèquement asynchrone,

géré par des messages/événements (ou lecture/écriture en base) en raison de la complexité des traitements (anonymisation, NER, prédiction).

3. Conception de Chaque Microservice

3.1. Diagramme de Classes : ProxyFHIR



Le diagramme de classes pour le microservice ProxyFHIR (Java/Spring Boot) détaille la structure interne pour la synchronisation des données et l'accès FHIR.

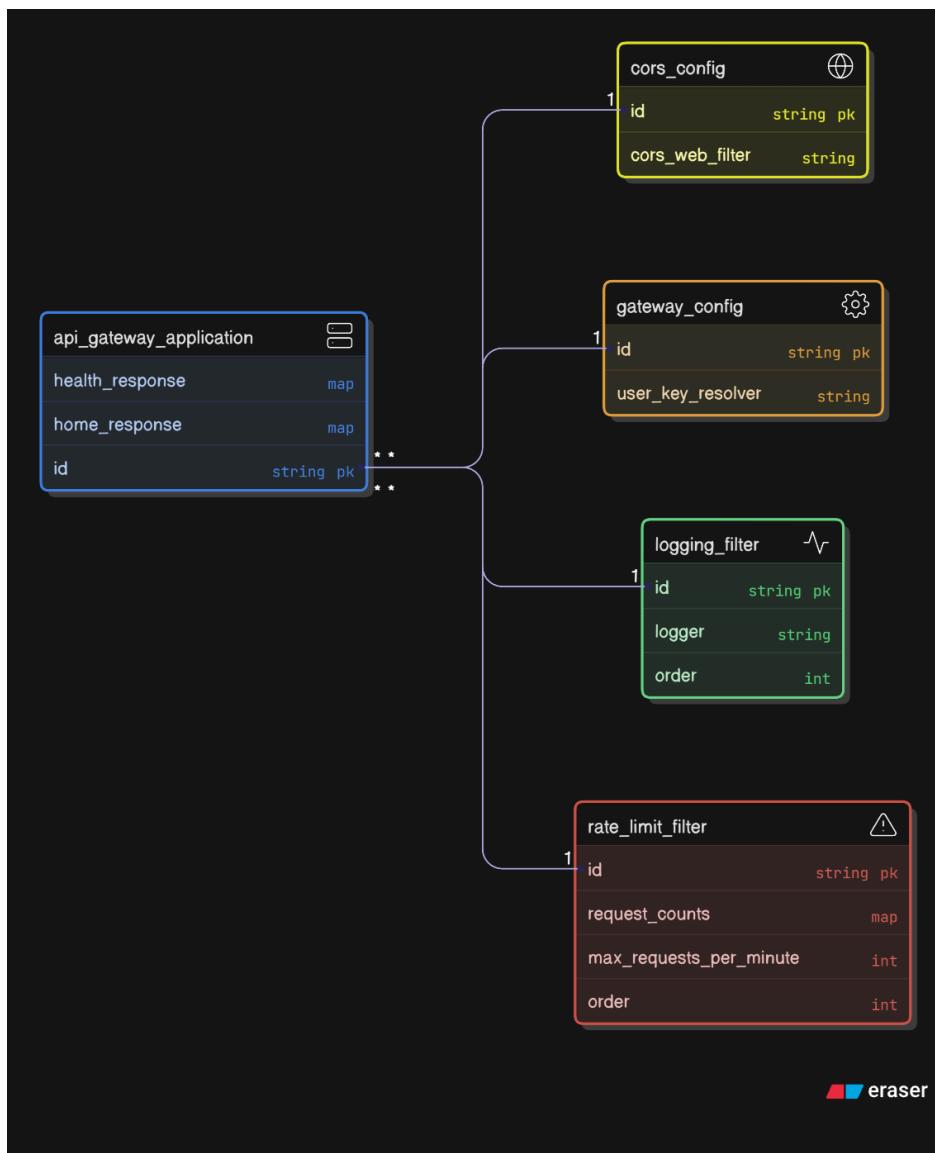
Classes Clés :

- **proxy_fhir_application** : Le point d'entrée de l'application Spring Boot.
- **fhir_sync_service** : Le cœur de la logique de synchronisation, orchestrant la récupération,

la validation et la sauvegarde des ressources FHIR.

- `fhir_proxy_controller` : Expose l'API REST pour les appels de synchronisation (e.g., `/fhir/sync/bulk`).
- `fhir_client_service` : S'occupe de la communication avec le serveur FHIR externe en utilisant la librairie HAPI FHIR.
- `fhir_resource_repository / fhir_bundle_repository` : Composants d'accès aux données (DAO) pour stocker les ressources et les Bundles dans la base de données (via Spring Data JPA).
- `fhir_config / database_config` : Classes de configuration pour les connexions.

3.2. Diagramme de Classes : API Gateway



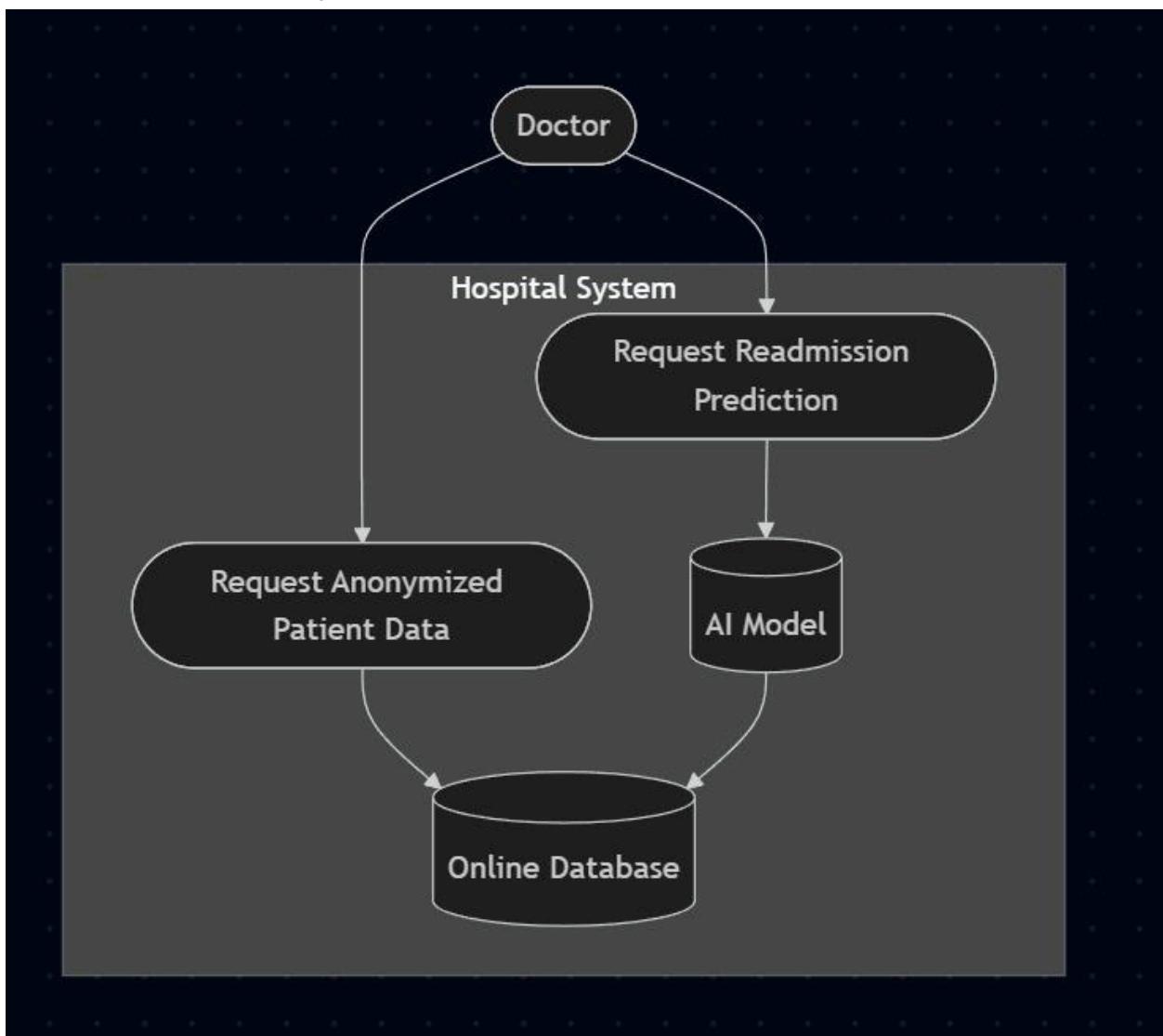
Le diagramme de classes pour l'API Gateway montre sa structure pour la gestion des requêtes, des configurations, et des filtres de sécurité/monitoring.

Classes Clés :

- api_gateway_application : Le point d'entrée de l'API Gateway.
- gateway_config : Gère la configuration de base de la passerelle, incluant la résolution de la clé utilisateur (user_key_resolver).
- cors_config : Gère la configuration CORS via un filtre web.
- logging_filter : Un filtre qui gère l'enregistrement des requêtes et des réponses, avec un ordre d'exécution (order).
- rate_limit_filter : Un filtre pour la limitation du taux de requêtes (max_requests_per_minute), assurant la résilience.

3.3. Cas d'Utilisation

Le cas d'utilisation principal est la **Prédiction du Risque de Réadmission** déclenchée par un acteur (le Médecin ou le système lui-même).



Acteurs et Cas d'Utilisation :

- **Acteur : Doctor (Médecin Traitant)**
 - **Request Readmission Prediction** : Consulter le score de risque d'un patient (via le Dashboard Web, qui appelle ScoreAPI).
 - **Request Anonymized Patient Data** : Accéder aux données anonymisées (indirectement, en consultant le dossier).
- **Acteur : Hospital System (Serveur FHIR)**
 - **Request Readmission Prediction** : Déclencher le processus de prédition à la sortie du patient.
- **Cas d'Utilisation :**
 - **Request Anonymized Patient Data** (Récupération des données FHIR, Anonymisation par DeID).
 - **AI Model** (Featurizer -> ModelRisque ->ScoreAPI) : Le système cœur effectuant l'extraction des features et la prédition.
- **Base de Données (Online Database)** : Stockage des données patients et des résultats de prédition.

4. Maquettes UI/UX réalisées avec Figma

Les maquettes fournies montrent une interface utilisateur professionnelle et orientée utilisateur pour le personnel médical.

4.1. Liste des Patients

Liste des patients

Consultez et gérez les dossiers des patients, filtrez par risque de réadmission et autres critères pour une vue d'ensemble rapide.

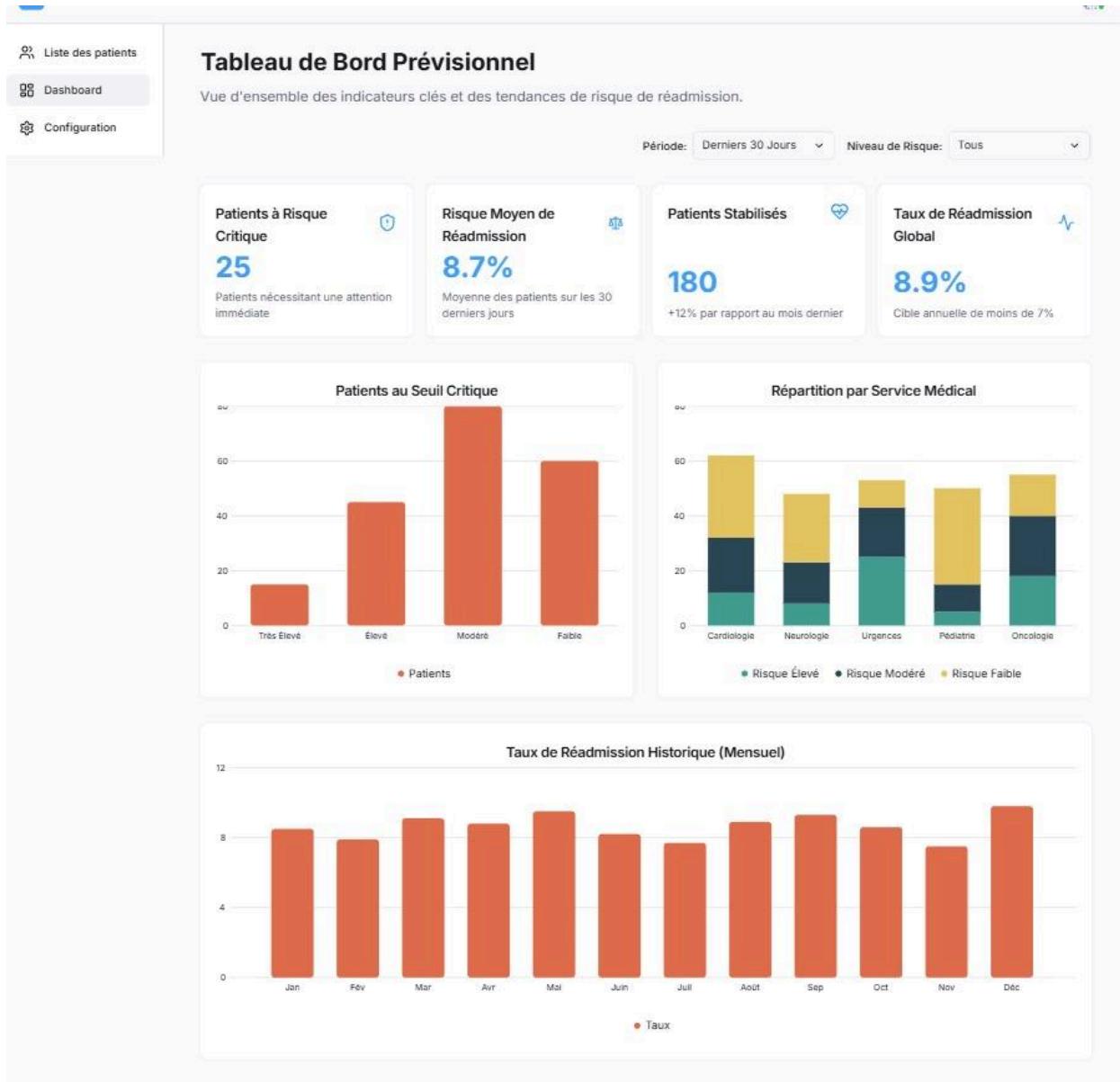
ID Patient	Nom Complet	Âge	Genre	Service	Dernier Diagnostic	Date de Consultation	Risque de Réadmission	Actions
P001	Jean Dupont	65	Homm e	Cardiologi e	Insuffisance cardiaque	2023-11-15	Très Élevé	Voir Détails >
P002	Marie Curie	72	Femme	Oncologie	Cancer du poumon	2023-10-20	Élevé	Voir Détails >
P003	Pierre Martin	50	Homm e	Neurologie	AVC	2023-12-01	Modéré	Voir Détails >
P004	Sophie Petit	34	Femme	Pédiatrie	Asthme chronique	2023-09-10	Faible	Voir Détails >
P005	Antoine Roy	80	Homm e	Gériatrie	Maladie d'Alzheimer	2024-01-05	Très Élevé	Voir Détails >

< Previous 1 2 3 Next >

© 2025 PrediCare. All rights reserved.

- **Fonctionnalité :** Affichage tabulaire des patients.
- **Informations Clés :** ID Patient, Nom Complet, Âge, Genre, Service, Dernier Diagnostic, Date de Consultation, et surtout le **Risque de Réadmission** (codé par couleur : Très Élevé, Élevé, Modéré, Faible).
- **Interaction :** Filtres de recherche (par ID/Nom, Risque, Service, Statut) et bouton "Voir Détails" pour accéder au dossier.

4.2. Tableau de Bord Prévisionnel



- Fonctionnalité :** Vue d'ensemble des indicateurs clés.
- Indicateurs :**
 - Patients à Risque Critique (e.g., 25).
 - Risque Moyen de Réadmission (e.g., 8.7%).
 - Patients Stabilisés (e.g., 180).
 - Taux de Réadmission Global (e.g., 8.9% vs. Cible annuelle de moins de 7%).
- Visualisations :** Graphiques de répartition des patients selon le **Seuil Critique** (Très Élevé à Faible) et la **Répartition par Service Médical** (Cardiologie, Neurologie, etc.). Un graphique historique mensuel du **Taux de Réadmission** est également présent.

4.3. Détail du Patient

The screenshot shows the PrediCare application's configuration interface. It features a sidebar with navigation links: 'Liste des patients', 'Dashboard', and 'Configuration' (which is selected). The main content area is divided into three sections:

- Section 1: Configuration des services**

Connexion à la base de données FHIR
Configurez les paramètres de connexion à votre serveur FHIR.

URL du serveur FHIR:
https://fhir.example.com/R4

Clé API (si applicable)

Endpoint OAuth 2.0 (si applicable)
https://auth.example.com/oauth2/token

Buttons: Tester la connexion, Enregistrer les modifications
- Section 2: Endpoints du microservice d'IA**

Définissez les URLs et les paramètres d'autentification pour le service de prédiction.

URL du service de prédiction:
https://ai.example.com/predict-readmission

Clé API du service IA

Format de requête/réponse:
JSON

Buttons: Tester le service, Enregistrer les modifications
- Section 3: Seuils critiques et alertes**

Configurez les pourcentages de risque de réadmission et les notifications d'alerte.

Seuil de risque critique (%): 70%

Activer les notifications d'alerte:

Destinataires d'e-mail (séparés par des virgules):
admin@example.com, urgences@example.com

URL du Webhook (pour intégration externe):
https://webhook.example.com/alerts

Buttons: Enregistrer les modifications

© 2025 PrediCare. All rights reserved.

- **Fonctionnalité :** Panneau de gestion des configurations système.
- **Configuration FHIR :** Connexion au serveur FHIR (URL, Clé API, Endpoint OAuth 2.0).
- **Configuration Microservice d'IA :** Endpoints de prédiction (/predict-readmission), Clé API, Format de requête/réponse (JSON).
- **Seuils Critiques et Alertes :**
 - Définition du **Seuil de risque critique** (e.g., 70%).

- Activation des notifications d'alerte.
- Destinataires d'e-mail et URL du Webhook pour l'intégration externe.